IN THE CLAIMS

Please enter the following amendments to the claims:

1. (Currently Amended) A method for morphing geometric shapes by using a direction map in order to smoothly morph more than two different geometric shapes, comprising the steps of:

a) extracting a direction map from each of the geometric shapes;

b1) rotating the direction maps according to an initial condition of merging;

b2) arranging direction vectors;

b3) merging the direction vectors to generate a merged direction map;

c) performing group-based scaling of the merged direction map; and

d) generating a polygon by using an inverse function of the merged direction map.

2. (Original) The method as recited in claim 1, further comprising the step of:

e) determining the number of in-betweens to be generated, and determining a group-based scaling method and a speed controlling method according to the number of in-betweens.

3. (Original) The method as recited in claim 1, further comprising the step of:

f) repeating said step c) and said step d) by changing a scaling factor.

4. (Previously Presented) The method as recited in claim 1, wherein the direction map is defined as a circular list of direction vectors and arranged based on an equation as:

$$d_i \times d_{i+1} = x_i y_{i+1} - y_i x_{i+1} \leq 0,$$

wherein d_i is an i^{th} direction vector and x_i and y_i are coordinates of d_i.

5. (Canceled)

6. (Currently Amended) The method as recited in claim 1, wherein if merged direction maps generates concave polygons, control shapes are decomposed into convex subsets and said steps b2) and b3) are executed.

7. (Currently Amended) The method as recited in claim 1, wherein if merged direction maps generates concave polygons, direction maps for convex hulls of control shapes are generated and said steps b2) to b3) are executed.

8. (Currently Amended) The method as recited in claim 1, wherein if merged direction maps are generating convex polygons, the direction maps are arranged to satisfy equations as:

$$d_l \times d_{l+1} = x_l y_{l+1} - y_l x_{l+1} \leq 0$$

$$d_l \times d_m = x_l y_m - y_l x_m \leq 0$$

$$d_m \times d_{l+1} = x_m y_{l+1} - y_m x_{l+1} \leq 0$$

$$d_l \times d_m = x_l y_m - y_l x_m > 0$$

$$d_m \times d_{l+1} = x_m y_{l+1} - y_m x_{l+1} > 0,$$

wherein an i^{th} direction vector of the direction map is described as d_i.

9. (Currently Amended) The method as recited in claim 1, wherein if every merged direction map does not generate a convex polygon, the direction maps are arranged to satisfy equations as:

$$d_l \times d_{l+1} = x_l y_{l+1} - y_l x_{l+1} > 0$$

$$d_l \times d_m = x_l y_m - y_l x_m \geq 0$$

$$d_m \times d_{l+1} = x_m y_{l+1} - y_m x_{l+1} \geq 0$$

$$d_l \times d_m = x_l y_m - y_l x_m < 0$$

$$d_m \times d_{l+1} = x_m y_{l+1} - y_m x_{l+1} < 0,$$

wherein an i^{th} direction vector of the direction map is described as d_i.

10. (Previously Presented) The method as recited in claim 1, wherein scaling of step (c) is interpolated by an equation as:

$$GS(D_{lm}, (S_l(t), S_m(t))),$$

$$S_l(t) = 1 - t,$$

$$S_m(t) = t,$$

wherein $S_i(t)$ is a list of scalar functions used for group-based scaling and D describes a direction map.

11. (Original) The method as recited in claim 10, wherein when N numbers of in-betweens are generated at a linear speed, the list of scalar functions expressed as below equations is used, herein the equations are:

$$(S_l(t), S_m(t)) = (1 - t, t) = (1 - t(n), t(n))$$

$$t(n) = \frac{n}{N+1}, (0 < n < N + 1).$$

12. (Original) The method as recited in claim 10, wherein when a speed of convergence is increased toward a control shape, the list of scalar functions expressed as below equations is used, herein the equations are:

$$(Sl(t), Sm(t)) = \frac{1}{s(1-t)+t}(s(1-t), t)$$

$$= \frac{1}{s(1-t)+t}(s(1-t(n)), t(n))$$

$$t(n) = \frac{n}{N+1}, \ (0 < n < N+1).$$

13. (Original) The method as recited in claim 1, wherein the inverse function of the direction map at said step d) satisfies equations as:

$DM^{-1}(D) = A$

$DM(A) = DM(DM^{-1}(D)) = D,$

wherein a function for mapping one polygon into a direction map is expressed as DM and a direction map is described as D.

14. (Original) The method as recited in claim 1, wherein at said step d), a polygon is generated based on equations as:

$$p_0 = p_{init}$$

$$p_{i+1} = p_i + d_i$$

$$p_{n0} = p_n + d_n$$

$$p_{i+m+1} = p_i + \sum_{j=i}^{i+m} d_j,$$

wherein an i^{th} vertex of a polygon is described as p_i and an i^{th} direction vector of the direction map is described as d_i.

15. (Original) The method as recited in claim 1, wherein at said step d), a polygon is generated by subsequently generating in-betweens based on an equation as:

$A(t) = DM^{-1}(D(t)),$

wherein $A(t)$ is an in-between and t is a scaling factor,

wherein an inverse function for mapping a direction map into one polygon is expressed as DM^{-1} and a direction map is described as D.

16. (Original) The method as recited in claim 15, wherein the in-betweens are generated by using Minkowski operation expressed as:

$$\begin{aligned}(1-t)A_0 \oplus tA_1 &= DM^{-1}(GS((DM(A_0) + DM(A_1)),(1-t,t))) \\ &= DM^{-1}(GS(D_0) + D_1,(1-t,t))) \\ &= DM^{-1}(GS(D,(1-t,t))) \\ &= DM^{-1}(\overline{D(t)}) \\ &= A(t).\end{aligned}$$

17. (Previously Presented) The method as recited in claim 15, wherein when a plurality of geometric shapes are morphed, a polygon is generated by Bezier control method based on equations as:

$$A(t) = DM^{-1}(SG(\underline{D},(B_0^M, \ldots, B_M^M(t)))),$$

$$\underline{D} = \sum_{i=0}^{M} DM(A_i) = \sum_{i=0}^{m} D_i,$$

$$B_i^M(t) = \binom{M}{i} t^i (1-t)^{M-i}.$$

18. (Original) The method as recited in claim 15, wherein when a plurality of geometric shapes are morphed, a polygon is generated by Blossom control method based on an equation as:

$$\begin{aligned}A(t_1,t_2,t_3) &= A_0^3[t_1,t_2,t_3] \\ &= DM^{-1}(SG(\overline{D},(B_0^3(t_1,t_2,t_3), \ldots, B_3^3(t_1,t_2,t_3)))),\end{aligned}$$

wherein a 3rd order base function of Blossom is described as $B_0^3(t_1,t_2,t_3)$ and a 3rd order Blossom is described as $A_0^3[t_1,t_2,t_3]$.

19. (Original) The method as recited in claim 1, further comprising the step of:

e) verifying the in-betweens generated by geometric shape morphing.

20. (Original) The method as recited in claim 19, wherein the generated in-betweens are verified by the normalized direction map based on an equation as:

$$\begin{aligned} N(DM(A(t))) &= N(DM(DM^{-1}(DA(t)))) \\ &= N(D(t)) \\ &= N(\overline{D}), \end{aligned}$$

wherein N is a normalizing function and a function for mapping one polygon into a direction map is expressed as DM,

wherein a direction map is described as D and an inverse function for mapping a direction map into one polygon is expressed as DM^{-1},

wherein a polygon is described as A.

21. (Original) The method as recited in claim 19, wherein the generated in-betweens are verified by the number of vertices based on an equation as:

$$N_{lm} = N_l + N_m - N_c,$$

wherein the numbers of vertices of control shapes A_l and A_m are N_l and N_m, and the number of line segments that have the same direction is N_c, the number of vertices of the generated polygon is N_{lm}.

22. (Previously Presented) The method as recited in claim 19, wherein the generated in-betweens are verified by the length of boundary line based on equations as:

$$A_{lm} = DM^{-1}(GS((DM(A_l) + DM(A_m)),(1 - t,t))),$$

$$L_{lm} = (1 - t)L_l + tL_m,$$

wherein when the boundary lengths of control shapes A_l and A_m are L_l and L_m, L_l and L_m are interpolated by the boundary lengths L_{lm} of the generated polygon A_{lm} by merging two control shapes.

23. (Currently Amended) A computer readable recording medium for storing a program that implements a geometric shape morphing method for morphing at least two different geometric shapes by using direction maps in a morphing apparatus including a microprocessor, the geometric shape morphing method comprising the steps of:

a) extracting a direction map from each of the geometric shapes;

b1) rotating the direction maps according to an initial condition of merging;

b2) arranging direction vectors;

b3) merging the direction vectors to generate a merged direction map;

c) performing group-based scaling of the merged direction map; and

d) generating a polygon by using an inverse function of the merged direction map.